DOCUMENT RESUME

ED 425 735 IR 019 223

AUTHOR	Thibeault, Nancy
TITLE	Web-Based Campus Directory.
PUB DATE	1998-00-00
NOTE	9p.; In: Association of Small Computer Users in Education: Proceedings of the ASCUE Summer Conference (31st, North Myrtle Beach, SC, June 7-11, 1998); see IR 019 201.
PUB TYPE	Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Access to Information; College Faculty; *Computer Interfaces; *Computer Software Development; Computer Uses in Education; *Databases; *Directories; Higher Education; *Online Systems; *World Wide Web
IDENTIFIERS	Miami University OH; Web Sites

ABSTRACT

Each semester at Miami University (Ohio) the secretaries struggle to update the Campus Directory database in order to print the campus telephone and office/class hours directory. It is a time-consuming process and the directory is usually late and out-of-date by the time it is printed. The university's Computer Services Manager has created a Web interface to their existing database, and now the directory is available on the Web. The information in the Web directory is current, and the secretaries' workload has been reduced. Faculty and staff members can print a copy of the latest information as needed. In the new system, faculty and staff members can update their database information using a Web form. The Web directory provides immediate access to the updated information. Each time the Web directory is accessed, the database is queried and the latest information is displayed. The Web directory consists of the telephone listing with a link from each entry to a pop up window containing a picture of the faculty/staff member, along with his/her current class/office hour information. This paper explains the files used to generate the Web-based directory. (AEF)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

Web-Based Campus Directory

PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

C.P. Singer

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC).

Nancy Thibeault
Computer Services Manager
Miami University
Middletown Campus
4200 E. University Blvd.
Middletown, OH 45042
(513) 727-3355

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

ED 425 735

Abstract

Each semester, our secretaries struggle to update the Campus Directory database in order to print the campus telephone and office/class hours directory. This is a time consuming process and the directory is usually late and out of date by the time it is printed. I have solved these problems by creating a Web interface to their existing database. Now the directory is available on the Web. The information in the Web directory is current and our secretaries' workload has been reduced. Faculty and staff members can print a copy of the latest information as needed.

In the new system, faculty and staff members can update their database information using a Web form. The Web directory provides immediate access to the updated information. Each time the Web directory is accessed, the database is queried and the latest information is displayed.

The Web directory consists of the telephone listing with a link from each entry to a pop up window containing a picture of the faculty/staff member along with their current class/office hour informatio

This paper explains the files used to generate the Web-based directory.

Microsoft Visual InterDev

The Web interface was developed using Microsoft Visual InterDev. Visual InterDev is an integrated development environment that provides all the tools you need to develop and manage a sophisticated Web application. It includes integrated programming, database development, site management and content editing tools. The Visual InterDev system requirements are shhown in Table 1.

Visual InterDev is a tool for developing Active Server Pages (ASP). ASP files are HTML files that include scripts that are processed on a Microsoft Windows NT/95 server running Internet Information Server (IIS) with Active Server Pages Extensions. ASP files include VBScript code enclosed in <% %> tags, which is converted to HTML by the server before being sent to the browser. The result is a pure HTML page, which is browser and platform independent.

Visual InterDev provides a complete development environment for creating Active Server applications, including tools that automatically generate much of the script. The tools generate the code to establish the database connection, perform database queries and integrate the data into dynamically generated HTML pages.

IR019223

System Requirements	Server	Developer	Client
Hardware	Minimum: 486/66 w/16MB RAM Recommended: Pentium w/ 32MB RAM	Minimum: 486/66 w/16MB RAM Recommended: Pentium w/ 32MB RAM	Any computer capable of running a graphical Web browser
Operating System	Windows NT (preferred) Windows 95	Windows NT Windows 95	Any
Software	IIS Active Server Extensions FrontPage Extensions	Visual InterDev	Web browser such as Netscape or Internet Explorer

Table 1. Visual InterDev Systme Requirements

The Database Tables

The Directory Database is a Microsoft Access database that is used by the Campus Administrator's office to produce the campus directory and generate mailing labels, email lists, contracts, etc. The Web interface uses two of the existing database tables: Directory and Class_Hours, shown in Table 2. The two tables are linked on the EmployeeID field. The directory table contains an entry for each faculty/staff member. The Class_Hours table includes office and class hour entries for each faculty member for every time slot. The Type field indicates whether the particular time is an office hour or a class hour. If it is a class hour, then the Course field contains the abbreviation for the course.

Directory	Class_Hours
EmployeeID	EmployeeID
LastName	Type
FirstName	Course
Prefix	StartTime
EmailName	EndTime
DeptArea	Mon
Office_no	Tue
Building	Wed
Extension	Thu
	Fri

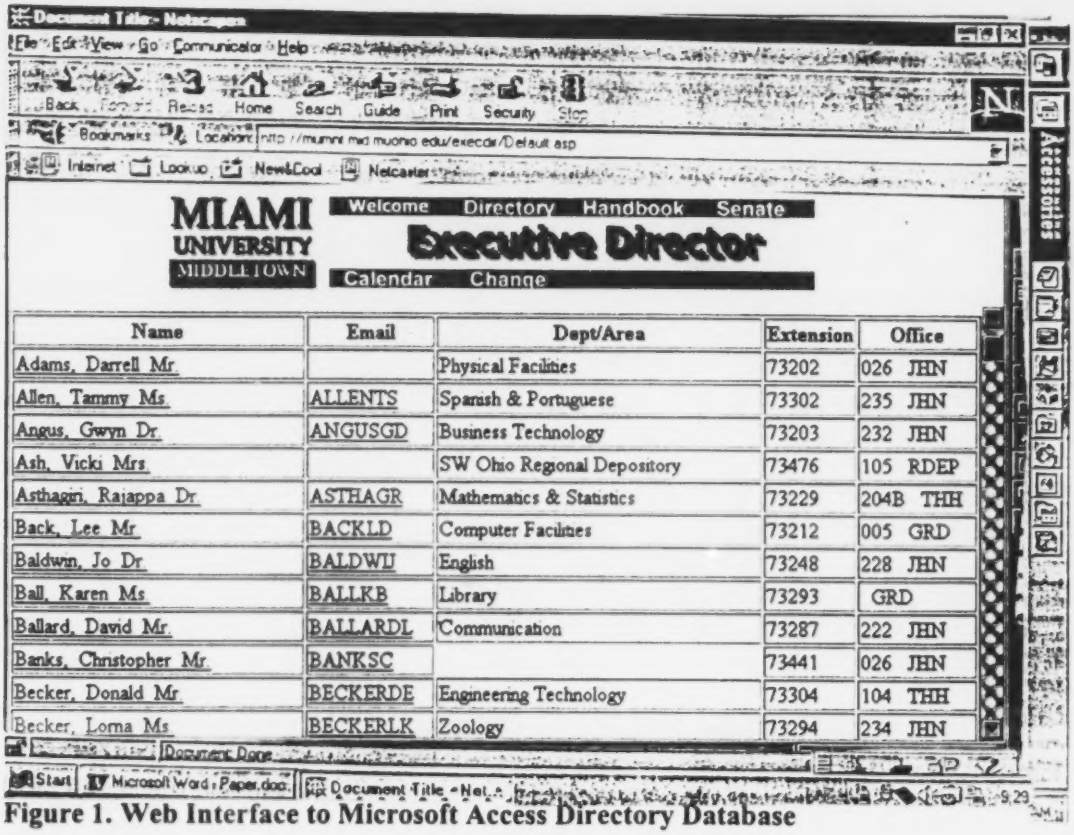
Table 2. Microsoft Access Directory Database Tables

The Web Application

The Web interface to the Directory Database is shown in Figure 1. Each name is a hyperlink to a pop-up window (see Figure 4) containing the faculty/staff member's picture, directory data, and class and office hours information. The Email Name is a mailto link that allows the user to send email to a faculty/staff member simply by clicking on the Email Name.

The Web interface is generated using two ASP files: Table.ASP and OfficeHrs.ASP. Table.ASP consists of a main program that makes the database connection, executes a query on the database then passes the resulting recordset to subroutine GenerateTable which produces the HTML table shown in Figure 1. When a user clicks on a hyperlinked name, the javaScript function openWin is called. The EmployeeID for that name is passed to the openWin function, which in turn calls OfficeHrs.ASP to populate the Pop-Up window shown in Figure 4.

Table.ASP ⟶ GenerateTable ⟶ openWin ⟶ OfficeHrs.ASP



Name	Email	Dept/Area	Extension	Office
Adams, Darrell Mr		Physical Facilities	73202	026 JHN
Allen, Tammy Ms	ALLENTS	Spanish & Portuguese	73302	235 JHN
Angus, Gwyn Dr	ANGUSGD	Business Technology	73203	232 JHN
Ash, Vicki Mrs		SW Ohio Regional Depository	73476	105 RDEP
Asthagiri, Rajappa Dr	ASTHAGR	Mathematics & Statistics	73229	204B THH
Back, Lee Mr	BACKLD	Computer Facilities	73212	005 GRD
Baldwin, Jo Dr	BALDWIJ	English	73248	228 JHN
Ball, Karen Ms	BALLKB	Library	73293	GRD
Ballard, David Mr	BALLARDL	Communication	73287	222 JHN
Banks, Christopher Mr	BANKSC		73441	026 JHN
Becker, Donald Mr	BECKERDE	Engineering Technology	73304	104 THH
Becker, Lorna Ms	BECKERLK	Zoology	73294	234 JHN

Figure 1. Web Interface to Microsoft Access Directory Database

Table.ASP

Table.ASP generates the Web interface shown in Figure 1. The "main program" ASP code is shown in Figure 2. It creates and opens a connection to the Microsoft Access Directory database, then sends an instruction to the database indicating the name of the database query to be executed. The resultant data is returned to the recordset variable rs. Subroutine GenerateTable is then called to generate the HTML page using the data from the recordset parameter rs.

BEST COPY AVAILABLE

BEST COPY AVAILABLE

```
<%set conn = Server.CreateObject("ADODB.Connection")
set cmd = Server.CreateObject("ADODB.Command")
Conn.Open Session("ExDirConn_ConnectionString")
cmd.ActiveConnection = conn
cmd.CommandText = "Nancy_query"
set rs = cmd.execute
GenerateTable rs%>
```

Figure 2. Table.ASP Main Program

GenerateTable

The GenerateTable subroutine shown in Figure 3 produces the HTML table. Note how the ASP code is interspersed with the HTML tags. The ASP Extensions of the IIS Web server will execute the ASP script, replacing all the rs variables with the actual data.

The GenerateTable subroutine first creates the Table and Headings from the HTML tags. Each pass through the while loop, a record from the recordset is used to produce a table row.

```
<% Sub GenerateTable( rs )%>
<TABLE BORDER=1>
<TR><TH>Name<BR></TH><TH>Email</TH><TH>Dept/Area</TH><TH>Extension</TH>
<TH>Office</TH></TR>
 <%  while not rs.EOF%>
   <TR>
  <td><A HREF="javascript: openWin(<%=rs(0)%>)"> <%=rs("LastName")%>,
           <%=rs("Firstname")%>  <%=rs("Prefix")%></A></td>
          <% if isNull(rs("EmailName")) then%>
                  <td> </td>
          <%else%>
                  <TD>
                  <A HREF="mailto:<%=rs("EmailName")%>@MUOhio.edu">
                  <%=rs("EmailName")%></A>
                  </TD>
          <%end if%>
          <TD><%=rs("DeptArea")%></TD>
          <TD><%=rs("Extension")%></TD>
          <TD><%=rs("office_no")%>   <%=rs("Building")%></TD>
          </TR>
   <%rs.MoveNext
  wend
  Response.Write( "</TABLE>" )
End Sub%>
```

Figure 3. Subroutine to Generate HTML for Campus Directory

The LastName, FirstName becomes a hyperlink to a javaScript function that opens a pop-up window. When a user clicks on a name a new window is opened displaying the details for that individual as shown in Figure 4.

When the code is executed on the server, each of the <%=rs(x)%> references is replaced with the actual data in the current record of the recordset.

For example, the record

EmpID	LastName	FirstName	Prefix	EmailName
1796	Hurn	Janet	Ms.	HrunJE

will replace the rs variables in the following code segment:

```
<A HREF="javascript: openWin(<%=rs(0)%>)">
<%=rs("LastName")%>,  <%=rs("Firstname")%>  <%=rs("Prefix")%></A>
```

generating the following HTML:

```
<A HREF="javascript: openWin(1796)"> Hurn,   Janet   Ms.</A>
```

A user can simply click on the Email Name in the table to send email to any faculty/staff member.

The Email Name is a mailto link formed by concatenating the EmailName from the recordset with @MUOhio.Edu.

For example:

```
<A HREF="mailto:<%=rs("EmailName")%>@MUOhio.edu">
```

is replaced with

```
<A HREF="mailto:HurnJE@MUOhio.edu">
```

The table row is completed using the remaining data from the recordset. This one subroutine writes the HTML for the entire directory table.

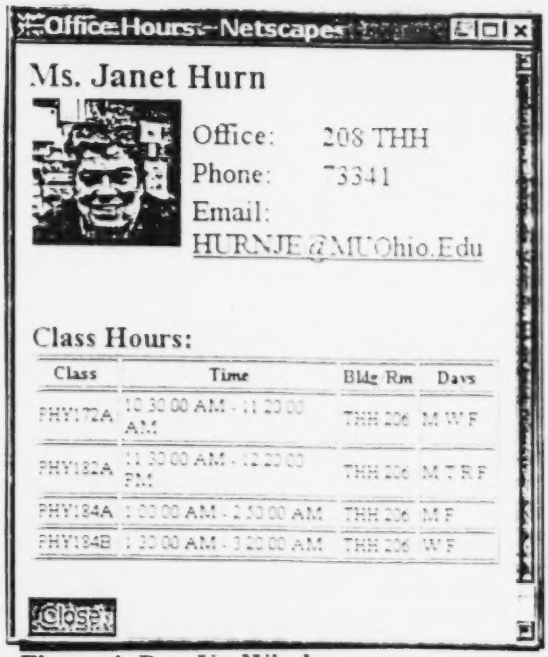


Figure 4. Pop-Up Window

openWin

The javaScript function shown in Figure 5 creates a new browser window which is populated by OfficeHrs.ASP. The resultant window is shown in Figure 4.

"OfficeHrs.asp?EmpID="+EmpID sends a request to the server for the OfficeHrs.ASP page passing the EmpID as a query string parameter. OfficeHrs.ASP then uses the EmpID to select the data for that faculty/staff member.

```
function openWin (EmpID)
        {
        var newWin = window.open("OfficeHrs.asp?EmpID=" + EmpID,
        OfficeHrsWin","toolbar=no,location=no,directories=no,status=no,
        scrollbars=yes, resizable=yes,copyhistory=no,width=350,height=400")
        }
```

Figure 5. JavaScript Function to open Pop-Up Window

OfficeHrs.ASP

The ASP code shown in Figure 6 displays the faculty/staff member's name, picture, office location, email address and phone extension. The ASP code makes the database connection, then uses the EmpID passed in as a query string parameter to create the Select query. The query is executed against the database and the resulting information is displayed in the window. Notice how concatenating the EmailName with the .jpg extension forms the file name for the picture.

<img src="images/<%=rs("EmailName")%>.jpg">

If the EmailName in the current record of rs is HurnJE, then <%=rs("EmailName")%> will be replaced by HurnJE resulting in

<img src="images/HurnJE.jpg">

```
<%set conn = Server.CreateObject("ADODB.Connection")
set cmd = Server.CreateObject("ADODB.Command")
Conn.Open Session("ExDirConn_ConnectionString")
cmd.ActiveConnection = conn
cmd.CommandText = "Select Prefix, LastName, FirstName, EmailName, Extension, Office_no, Building  from Directory
where EmployeeID = " & request.querystring("EmpID")
set rs = cmd.execute%>
<b><Font Size=+1>
<%=rs("Prefix")%> <%=rs("FirstName")%> <%=rs("LastName")%></Font></b><BR>
<table border=0>
<tr><td>
<img src="images/<%=rs("EmailName")%>.jpg" Width=100 Height=100
ALT="<%=rs("Prefix")%> <%=rs("FirstName")%> <%=rs("LastName")%>">
<BR><BR></td>
<td>
<table border=0>
<tr><td>Office:</td><td> <%=rs("Office_no")%> <%=rs("Building")%></td></tr>
<tr><td>Phone:</td><td> <%=rs("Extension")%></td></tr>
<%if not isNull (rs("EmailName")) then%>
        <tr>
        <td colspan=2>Email: <BR>
        <A Href="mailto:<%=rs("EmailName")%>@MUOhio.Edu"><%=rs("EmailName")%>
        @MUOhio.Edu</A></td></tr>
<%end if%></table></td></tr>
</table>
```

Figure 6. OficeHrs.ASP code to display picture, name, office, email and phone number

BEST COPY AVAILABLE

Figure 7 shows the ASP code used to generate the Office Hours table in the pop-up window. A query is executed to retrieve the office hours for the selected EmployeeID then displays the table shown at the bottom of Figure 4.

```
<%cmd.CommandText = "Select * from Class_Hours where EmployeeID =
" &request.querystring("EmpID")& " AND Type = 'Office'"
set rs1 = cmd.execute%>
<table border=0>
<tr>
<%if not rs1.eof and not rs1.bof then%>

        <td><B>Office Hours:</B><BR>
        <table border=1>
        <TR><TH><Font Size="1">Time</font></TH><TH>
        <Font Size="1">Days</font></TH></TR>
        <%while not rs1.eof%>
        <tr><td><Font Size="1">
        <%=rs1("StartTime")%> - <BR><%=rs1("EndTime")%></font></td>
        <td><Font Size="1">
        <%val=rs1("Mon") if not val = 0 then%>M<% end if%>
        <%val=rs1("Tue") if not val = 0 then%>T<% end if%>
        <%val=rs1("Wed") if not val = 0 then%>W<% end if%>
        <%val=rs1("Thu") if not val = 0 then%>R<% end if%>
        <%val=rs1("Fri") if not isnull(val) then%>F<% end if%>
        </font></td></tr>
        <%rs1.MoveNext%>
        <%wend%>
        </table>
        </td>
<%end if%>
```

Figure 7. OfficeHrs.ASP code to generate office hours table

The query retrieves all office hour records for the selected EmployeeID. The resultant recordset is stored in rs1. Each record of rs1 is then displayed in table format.

Summary

Visual InterDev is an excellent tool for developing database driven Web applications. It contains all the tools needed to create and test your application in a single integrated environment; however, it has a fairly steep learning curve. The wizards and visual tools are great for generating basic applications, but I found that you must understand HTML and ASP in order to fine-tune and create customized applications. The wizards generate too much unnecessary code that makes the applications difficult to understand and debug. I prefer to write my own code. It is a great development environment, but you need a background in HTML, programming and databases in order to use it effectively.

References

Teach Yourself Active Web Database Programming in 21 Days, Sams Net, 1997. Fleet, Warren, Chen, and Stojanovic.

Professional Active Server Pages, Wrox Press, 1997. Alex Homer, etal.

Special Edition Using Visual InterDev, Que, 1997. Mike Morrison

Inside Microsoft Visual InterDev, Microsoft Press, 1997. Ken Miller, Kenneth Spencer, and Eric Vincent.

Microsoft Visual InterDev Web Site. www.microsoft.com/vinterdev.